****

Class: SY-MCA   **Shift / Div:** A   **Batch:** S2   **Roll Number:** 52043

Name: Vanessa Reetu Prashant More   **Assignment No:** 2   **Date of Implementation:** 8. 9. 23

****

Q1. Prepare test plan for Online shopping system using MS word.

PES MCOE

Project Name	Online Shopping System
Product Name	Shopping System
Product Release Version	3.0

System Test Plan
Document version: 2.24

Date: 23/09/2023
Prepared By: Vanessa More
Reviewed By: Tanima Mandal

INDEX

<div align="center">**Online Shopping System (OSS)**</div>

1. Unique id/test plan identifier
Test plan identifier: OSS_PESMCOE_2023

2. Introduction
The Online Shopping System is in the process of being designed for our customer, PES MCOE. This Online Shopping System is an essential element of contemporary trade, affording consumers the ease of buying goods and services from their residences. The customer anticipates that the platform will be easy to use, safe, and dependable.

Objectives
The objective of this test plan is to define the testing approach and strategy for the Online Shopping system to ensure that it meets its functional and non-functional requirements.

References
- Online Shopping System Requirements Document, Version 1.25
- Online Shopping System Design Document, Version 1.25
- OSS_Project_Plan - OSS_23
- OSS Master Test Plan - OSS_PES_23

3. Test Items
- Login - OSS_1
- Search Items - OSS_2
- Add the items into the cart - OSS_3
- Add delivery address - OSS_4
- Confirm order - OSS_5
- Select payment mode - OSS_6
- Pay the bill - OSS_7
- Help - OSS_8

4. Features to Be Tested
- User login
- Product Search and Browsing
- Adding Products to the Cart
- Shopping Cart Management
- Order Placement
- Payment Processing
- User Account Management
- Product Reviews and Ratings
- Product Recommendations

5. Features Not to Be Tested

- Third-party services (e.g., payment gateways) - Will be tested separately by their respective providers.

6. Approach

The OSS will be tested thoroughly on several levels. The test cases, design specifications, test procedure as well as any test scripts are prepared & will be saved in test repository for using further releases.

All the events observed during test execution will be recorded in a test law that will be associated with each test. All failures will assign a severity level & will be recorded in the test incident report.

The system testing will be monitored through weekly test status meetings & the summary reports will be prepared by our technical lead Nidhi.

The following tools will be used to support the system testing efforts:

- Content management system
- Customer Relationship Management (CRM) Software
- Analytics Tools
- Search Engine Optimization (SEO) Tools
- Social Proof and Reviews Platforms

Testing types:

Functional Testing:

- User Interface (UI) Testing: Ensures that the website or app has a user-friendly interface, and all elements are displayed correctly.
- User Experience (UX) Testing: Focuses on the overall user experience, including navigation, ease of use, and responsiveness.
- Navigation Testing: Verifies that users can easily navigate through different sections of the site and access product categories and pages.
- Search Functionality Testing: Checks the search feature for accuracy in returning relevant results.

Compatibility Testing:

- Browser Compatibility Testing: Ensures that the website functions correctly on various web browsers (e.g., Chrome, Firefox, Safari, Edge).
- Device Compatibility Testing: Tests the compatibility of the system on different devices (desktops, tablets, and smartphones) and operating systems (iOS, Android).

Performance Testing:

- Load Testing: Determines how the system performs under heavy traffic loads to ensure it can handle peak usage without crashing.
- Stress Testing: Evaluates how the system behaves under extreme conditions, such as a sudden surge in traffic or simultaneous transactions.
- Speed and Response Time Testing: Measures the loading speed of web pages and responsiveness of the system.

Security Testing:

- Authentication and Authorization Testing: Verifies that user authentication and authorization mechanisms are secure and effective.
- Data Security Testing: Ensures the protection of user data, including encryption and secure storage.
- Payment Security Testing: Validates the security of payment processing and handling of sensitive payment information.

Usability Testing:

- User Testing: Involves real users interacting with the system to gather feedback on usability, user satisfaction, and any pain points.
- Accessibility Testing: Ensures that the website is accessible to users with disabilities, meeting WCAG guidelines.

Regression Testing:

- Ensures that new updates, features, or bug fixes do not negatively impact existing functionality.

Database Testing:

- Checks data integrity, database performance, and data retrieval accuracy.

7. Pass/Fail Criteria

Company's test policy & procedure document TM_2.0_23 describes a scale of severity levels for faults & failures. The scale ranges in values from 1-4 where 1 is failure that has a severe effect on the system & users. The value 4 indicates minimal effect. For the OSS, a test will be considered as pass if the value is between 3-4.Minor defects will be tracked and reported for resolution.

8. Suspend/Resume Criteria

Testing activities may be suspended if critical defects severely affect the testing environment. Testing will resume when these issues are resolved.

9. Test Deliverables

- Test Cases and Test Data
- Test Execution Reports
- Defect Reports
- Final Test Summary Report

10. Testing Tasks

- Test case design and documentation
- Test data preparation
- Test environment setup
- Test execution and defect reporting
- Test summary and reporting

11. Test Environment Hardware Requirements

- Operating System compatibility
- Web Servers (minimum of 2)
- Database Servers
- Desktop and Mobile Devices for Testing
- Internet Connectivity

12. Responsibilities

The test group members responsible for OSS are:

- Riya Kale,Testing Manager: Designing, executing, and reporting on test cases.
- Mihir Deshpande, Development Team Manager: Resolving defects and issues.
- Ronak Bafna, Project Manager: Overall test management and coordination.

13. Staffing and Training

All the testing staff had training in development of test design, test procedures, etc. They have the necessary training in test tool usage & also have experience in test planning, so no additional training for this project is required.

14. Scheduling

Sr. no.	Task	Responsibility	Designation	Start date	End date
1.	Test case design and documentation	Riya Kale	Testing Manager	10/4/23	14/4/23
2.	Test data preparation	Riya Kale	Testing Manager	18/4/23	25/4/23
3.	Test environment setup	Mihir Deshpande	Development Team Manager	1/5/23	10/5/23
4.	Test execution and defect reporting	Riya Kale	Testing Manager	15/5/23	28/5/23
5.	Test summary and reporting	Ronak Bafna	Project Manager	2/6/23	19/6/23

15. Risks and Contingencies

Risk: Delay in development may impact the testing schedule.
Contingency: Adjust the testing schedule accordingly.

Risk: Critical defects found during testing may require significant development efforts.
Contingency: Prioritise defect resolution and retesting.

16. Approvals

- Ms. Kiran Kumar
  (Test Manager)

  Date: 15/8/23   Signature:

- Mr. Mahesh Saaho
  (Software Quality Assurance Manager)

  Date: 19/8/23   Signature:

- Mr. Ronak Bafna
  (Project Manager)

  Date: 29/8/23   Signature:

_____

Progressive Education Society's

Modern College of Engineering, Pune

MCA Department

A.Y.2023-24

(410907) Software Testing Laboratory

****

Class: SY-MCA   **Shift / Div:** A   **Batch:** S2   **Roll Number:** 52043

Name: Vanessa Reetu Prashant More   **Assignment No:** 2   **Date of Implementation:** 8. 9. 23

****

Q2. Prepare a test plan for an identified Mobile Application - Banking App.

PES MCOE

Project Name	Vamo Banking App
Product Name	Shopping System
Product Release Version	3.0

System Test Plan
Document version: 3.25

Date: 23/09/2023
Prepared By: Vanessa More
Reviewed By: Tanima Mandal

INDEX

1. Unique id/test plan identifier
2. Introduction
3. Test items
4. Features to be tested
5. Features not to be tested
6. Approach
7. Pass/Fail criteria
8. Suspend/Resume criteria
9. Test deliverables
10. Testing tasks
11. Test environment hardware requirements
12. Responsibilities
13. Staffing and training bits
14. Scheduling
15. Risks and contingencies
16. Approvals

<p align="center">**Vamo Banking App (VBA)**</p>

1. Unique id/test plan identifier

Test plan identifier : VBA_MCOE_2023

2. Introduction

The Vamo Banking App is being developed for our client PES MCOE. This banking app emerges as a pioneering solution designed to empower our valued customers with unprecedented control over their financial lives. The client expects the system to be seamless, efficient, and secure.

Objectives

The objective of this test plan is to define the testing approach, scope, resources, and schedule for thoroughly testing the Mobile Banking Application. The goal is to ensure that the application is secure, functions as expected, and provides an excellent user experience while adhering to industry standards and regulations.

References

- Mobile Banking Application Requirements Document, Version 2.25
- Mobile Banking Application Design Document, Version 2.25
- Regulatory and Compliance Guidelines (e.g., banking industry regulations)
- VBA_Project_Plan - OSS_23
- VBA Master Test Plan - OSS_PES_23

3. Test Items

The test items include the Mobile Banking Application and its associated components, such as the user interface, features, and backend services.

4. Features to Be Tested

- User Authentication (Login and Registration)
- Account Balances and Transactions
- Fund Transfers (Internal and External)
- Bill Payments
- Mobile Check Deposits
- ATM and Branch Locator
- Notifications and Alerts
- Account Settings and Profile Management
- Security and Encryption

5. Features Not to Be Tested

● Third-party services (e.g., external payment processors) - These will be tested separately by their respective providers.

6. Approach

The VBA will be tested thoroughly on several levels. The test cases, design specifications, test procedure as well as any test scripts are prepared & will be saved in the test repository for using further releases.

All the events observed during test execution will be recorded in a test law that will be associated with each test. All failures will assign a severity level & will be recorded in the test incident report.

The system testing will be monitored through weekly test status meetings & the summary reports will be prepared by our technical lead Nidhi.

The following tools will be used to support the system testing efforts:

- User Analytics and Monitoring

- Authentication and Authorization Tools

- Fraud Detection Systems

- Mobile App Testing and Quality Assurance

- Database Management Systems (DBMS)

Testing types:

Functional Testing:
- Login and Authentication Testing: Verify that users can securely log in, reset passwords, and authenticate their identity.
- Account Management Testing: Test account creation, deletion, and profile updates.
- Transaction Testing: Validate money transfers, bill payments, and transaction history.
- ATM/Branch Locator Testing: Confirm that the app correctly displays the nearest ATMs and branches.

Security Testing:
- Encryption Testing: Ensure data transmitted between the app and servers is encrypted to protect user information.
- Authentication Testing: Verify that multi-factor authentication (MFA) and biometric authentication methods (e.g., fingerprint or facial recognition) work securely.
- Data Storage Testing: Ensure that sensitive data is stored securely on the device, using secure storage mechanisms.

Compatibility Testing:

- Device Compatibility Testing: Test the app on various mobile devices (iOS and Android) and screen sizes.
- Browser Compatibility Testing: If the app uses a web interface, ensure compatibility with different mobile browsers.
- Operating System Compatibility: Verify the app's compatibility with different OS versions (e.g., iOS 12, Android 10, etc.).

Regression Testing:

- Ensure that new updates or bug fixes do not negatively impact existing functionality.

User Acceptance Testing (UAT):

- Involve end-users in testing to ensure the app meets their needs and expectations.

7. Pass/Fail Criteria

Company's test policy & procedure document TM_4.25_23 describes a scale of severity levels for faults & failures. The scale ranges in values from 1-4 where 1 is failure that has a severe effect on the system & users. The value 4 indicates minimal effect. For the OSS, a test will be considered as pass if the value is between 3-4. Minor defects will be tracked and reported for resolution.

8. Suspend/Resume Criteria

Testing activities may be suspended if critical defects severely affect the testing environment. Testing will resume when these issues are resolved.

9. Test Deliverables

- Test Cases and Test Data
- Test Execution Reports
- Security Assessment Reports
- Accessibility Compliance Reports
- Defect Reports
- User Acceptance Test Results

10. Testing Tasks

- Test case design and documentation
- Test data preparation
- Test environment setup
- Test execution and defect reporting
- Security assessment and vulnerability scanning
- Accessibility testing
- User Acceptance Testing (UAT)
- Test summary and reporting

11. Test Environment Hardware Requirements

- Mobile devices (iOS and Android)
- Emulators and simulators for cross-platform testing
- Internet connectivity for backend testing
- Security testing tools

12. Responsibilities

The test group members responsible for OSS are:

- Bhushan Jain, Testing Team Manager: Designing, executing, & reporting on test cases.
- Ronak Bafna, Development Team Manager: Resolving defects and security vulnerabilities.
- Sakshi Methe, Compliance and Regulatory Team Manager: Ensuring adherence to industry standards and regulations.
- Kiran Kumar, Project Manager: Overall test management and coordination.

13. Staffing and Training

All the testing staff had training in development of test design, test procedures, etc. They have the necessary training in test tool usage & also have experience in test planning, so no additional training for this project is required.

14. Scheduling

Sr. no.	Task	Responsibility	Designation	Start date	End date
1.	Test case design and documentation	Bhushan Jain	Testing Manager	10/4/23	14/4/23
2.	Test data preparation	Bhushan Jain	Testing Manager	18/4/23	25/4/23
3.	Test environment setup	Ronak Bafna	Development Team Manager	1/5/23	10/5/23
4.	Test execution and defect reporting	Bhushan Jain	Testing Manager	15/5/23	28/5/23
5.	Security assessment and vulnerability scanning	Ronak Bafna	Development Team Manager	2/6/23	19/6/23
6.	Test summary and reporting	Kiran Kumar	Project Manager	22/6/23	4/7/23

15. Risks and Contingencies

Risk: Security breaches or data leaks during testing.
Contingency: Regularly update security protocols and monitor for vulnerabilities.

Risk: Unexpected changes in banking regulations.
Contingency: Maintain close communication with regulatory bodies and adapt as needed.

16. Approvals

- Ms. Minal Rao
  (Test Manager)

    Date: 15/8/23   Signature:

- Mr. Shruti Singh
  (Software Quality Assurance Manager)

    Date: 19/8/23   Signature:

- Mrs. Kiran Kumar
  (Project Manager)

    Date: 29/8/23   Signature:

****

Class: SY-MCA   **Shift / Div:** A   **Batch:** S2   **Roll Number:** 52043

Name: Vanessa Reetu Prashant More   **Assignment No:** 2   **Date of Implementation:** 8. 9. 23

****

Q3. Create manual Test cases for Online Shopping system using Excel.

.Test Cases for the Online Shopping System					
Module	Authentication	**Test Data Input**	User ID & Password	**Testing Date**	22-Sep
Test Scenario	Verification of the details for login page	**Tester**	Vanessa	**Authorized by**	Sakshi
Test case ID	**Test Case**	**Requirements**	**Expected Outcome**	**Actual Outcome**	**Status**
OSS_1_1	Valid login credetials	The user enters a valid user ID and password that exist in the system.	The user is successfully logged in to their account.	As expected	Pass
OSS_1_2	Invalid User ID	The user enters an invalid user ID that does not exist in the system.	An error message is displayed, indicating an invalid user ID.	As expected	Pass
OSS_1_3	Invalid Password	The user enters a valid user ID but an incorrect password.	An error message is displayed, indicating an incorrect password.	User is logged into their account	Fail
OSS_1_4	Empty User ID Field	The user leaves the user ID field empty.	An error message is displayed, indicating that the user ID field is required.	As expected	Pass
OSS_1_5	Empty Password Field	The user leaves the password field empty.	An error message is displayed, indicating that the password field is required.	As expected	Pass
OSS_1_6	Password Strength	The user enters a valid user ID and a weak password (e.g., less than 8 characters).	An error message is displayed, indicating that the password is weak.	Accepts the weak password entered by the user	Fail
OSS_1_7	User ID Case Sensitivity	The user enters a user ID with different letter case (e.g., "Username" instead of "username").	The system should be case-insensitive and process the login normally.	As expected	Pass
OSS_1_8	Password Case Sensitivity	The user enters a password with different letter case (e.g., "Password" instead of "password").	The system should be case-sensitive and display an error.	As expected	Pass

Module	Product search & browsing		Test Data Input	Search categories, product categories, etc.	Testing Date	28-Sep
Test Scenario	Verification of the details on product specification & browsing		Tester	Nidhi	Authorized by	Rajesh

Test case ID	Test Case	Requirements	Expected Outcome	Actual Outcome	Status
OSS_2_1	Basic Product Search	The user enters a valid search query in the search bar and presses "Enter" or clicks the search button.	A list of products matching the query is displayed. The user can see product names, prices, and images.	As expected	Pass
OSS_2_2	Empty Search Query	The user leaves the search query field empty and clicks the search button.	An error message or a prompt is displayed, indicating that a search query is required.	As expected	Pass
OSS_2_3	Searching with Filters	The user applies filters (e.g., price range, brand, size) to narrow down search results.	The displayed products are filtered according to the selected criteria.	The products are not filtered	Fail
OSS_2_4	Back Navigation	After browsing a product details page, the user clicks the "Back" button or uses the browser's back navigation.	The user is returned to the previous page (e.g., search results or category browsing) with their previous state intact.	As expected	Pass
OSS_2_5	Pagination	When there are more products than can be displayed on a single page, the user navigates to the next page.	The user sees a new set of products on the next page, with pagination controls for navigation.	Refreshes the page	Fail
OSS_2_6	Product Image Zoom	On the product details page, the user clicks on a product image.	The image is enlarged for better visibility, allowing the user to see product details more clearly.	Image is not enlarged	Fail

Module	Order placement & payment information		Test Data Input	Add items to cart, selection, etc.	Testing Date	28-Sep
Test Scenario	Verification of the details on the order placed		Tester	Tanay	Authorized by	Vijay

Test case ID	Test Case	Requirements	Expected Outcome	Actual Outcome	Status
OSS_3_1	Adding Items to Cart	The user adds one or more products to the shopping cart.	The selected products are added to the cart with correct quantities.	As expected	Pass
OSS_3_2	Cart Validation	The user reviews the shopping cart before proceeding to checkout.	The cart displays the selected items, quantities, prices, and a total order amount.	As expected	Pass
OSS_3_3	Cart Update	The user modifies the quantity of items in the cart (e.g., increases, decreases, or removes items).	The cart updates accordingly, showing the revised quantities and adjusted total amount.	The cart is not updated or revised	Fail
OSS_3_4	Out of Stock Product Handling	The user attempts to place an order for a product that is out of stock.	An error message is displayed, indicating that the product is out of stock, and the user cannot proceed.	As expected	Pass
OSS_3_5	Order Confirmation	The user confirms the order after reviewing the details.	An order confirmation page is displayed, showing the order number and expected delivery date.	As expected	Pass
OSS_3_6	Order Total Calculation	The user adds products to the cart with different prices and quantities, and the order total is calculated.	The order total accurately reflects the sum of the selected products, quantities, and any applicable taxes or discounts.	The order total is wrong	Fail
OSS_3_7	Payment Method Selection	The user selects a payment method (e.g., credit card, PayPal, cash on delivery) during checkout.	The selected payment method is displayed, and the user proceeds to enter payment details if required.	As expected	Pass
OSS_3_8	Payment Information Validation	The user enters incomplete or incorrect payment information during checkout.	An error message is displayed, indicating the missing or invalid fields, and the user cannot proceed.	As expected	Pass

Progressive Education Society's

Modern College of Engineering, Pune

MCA Department

A.Y.2023-24

(410907) Software Testing Laboratory

****

Class: SY-MCA   **Shift / Div:** A   **Batch:** S2   **Roll Number:** 52043

Name: Vanessa Reetu Prashant More   **Assignment No:** 2   **Date of Implementation:** 8. 9. 23

****

Q4. Create manual Test cases for Mobile Application - Banking App using Excel.

Test Cases for the Banking App					
Module	Authentication	**Test Data Input**	User ID & Password	**Testing Date**	5-Sep
Test Scenario	Verification of the details for login page	**Tester**	Tanisha	**Authorized by**	Rohit

Test case ID	Test Case	Requirements	Expected Outcome	Actual Outcome	Status
VBA_1_1	Valid login credetials	The user enters a valid user ID and password that exist in the system.	The user is successfully logged in to their account.	As expected	Pass
VBA_1_2	Invalid User ID	The user enters an invalid user ID that does not exist in the system.	An error message is displayed, indicating an invalid user ID.	As expected	Pass
VBA_1_3	Invalid Password	The user enters a valid user ID but an incorrect password.	An error message is displayed, indicating an incorrect password.	User is logged into their account	Fail
VBA_1_4	Empty User ID Field	The user leaves the user ID field empty.	An error message is displayed, indicating that the user ID field is required.	As expected	Pass
VBA_1_5	Empty Password Field	The user leaves the password field empty.	An error message is displayed, indicating that the password field is required.	As expected	Pass
VBA_1_6	Password Strength	The user enters a valid user ID and a weak password (e.g., less than 8 characters).	An error message is displayed, indicating that the password is weak.	Accepts the weak password entered by the user	Fail
VBA_1_7	User ID Case Sensitivity	The user enters a user ID with different letter case (e.g., "Username" instead of "username").	The system should be case-insensitive and process the login normally.	As expected	Pass
VBA_1_8	Password Case Sensitivity	The user enters a password with different letter case (e.g., "Password" instead of "password").	The system should be case-sensitive and display an error.	As expected	Pass

Module	Account balance & transactions		Test Data Input	Account no.,credentials, etc.	Testing Date	10-Sep
Test Scenario	Verification of the details on account		Tester	Karan	Authorized by	Nisha
Test case ID	Test Case	Requirements	Expected Outcome	Actual Outcome	Status	
VBA_2_1	Check Initial Balance	User has a valid account with a balance	Initial balance displayed	As expected	Pass	
VBA_2_2	Deposit Money	User wants to deposit money into their account	Account balance updated correctly	As expected	Pass	
VBA_2_3	Insufficient Funds	User attempts to withdraw more than the balance	Error message displayed	As expected	Pass	
VBA_2_4	Invalid Account	User enters an invalid account number	Error message displayed	As expected	Pass	
VBA_2_5	Transaction Limits	User tries to perform a transaction above limits	Transaction limit exceeded	As expected	Pass	
VBA_2_6	Account Lockout	User enters incorrect PIN multiple times	Account locked after attempts	Account is not locked	Fail	

Module	Account Settings & Profile Management		Test Data Input	Email, passwords, etc.	Testing Date	16-Sep
Test Scenario	Update profile with proper conditions		Tester	Vanessa	Authorized by	Vinit
Test case ID	Test Case	Requirements	Expected Outcome	Actual Outcome	Status	
VBA_3_1	User can change their password	User must be logged in	Password is successfully changed	Not allowing to change password	Fail	
VBA_3_2	User can link external accounts	User must be logged in	External accounts are successfully linked	As expected	Pass	
VBA_3_3	User can view recent login history	User must be logged in	Recent login history is displayed correctly	As expected	Pass	
VBA_3_4	User receives an email confirmation	User updates email address	An email confirmation is sent to the new email address	As expected	Pass	
VBA_3_5	User can recover their account	User enters correct account recovery information	Account recovery is successful	Account is not recovered	Fail	
VBA_3_6	User can select preferred language	User must be logged in	Preferred language is successfully updated	As expected	Pass	

****

Class: SY-MCA   **Shift / Div:** A   **Batch:** S2   **Roll Number:** 52043

Name: Vanessa Reetu Prashant More   **Assignment No:** 2   **Date of Implementation:** 29. 9. 23

****

Q1. Design template using Excel for Defect Report.

DEFECT REPOSITORY													
Project Name:													
Module Name:													
Release:													
Program Manager:													
Project Manager:													
Defect ID	**Date Created**	**Created By**	**Defect Description**	**Steps to reproduce the defects**	**Severity**	**Priority**	**Test case ID**	**Assigned To**	**Assigned Date**	**Status**	**Resolution Date**	**Expected Result**	**Actual Result**

****

Class: SY-MCA **Shift / Div:** A **Batch:** S2 **Roll Number:** 52043

Name: Vanessa Reetu Prashant More **Assignment No:** 2 **Date of Implementation:** 29. 9. 23

****

Q2. Prepare Defect Repository for Online shopping system using MS Excel

Project Name:	Online Shopping System
Module Name:	Authentication, Product search & browsing, Payment
Release:	2.4 version
Program Manager:	Vanessa More
Project Manager:	Kiran Shah

Defect ID	Date Created	Created By	Defect Description	Steps to reproduce the defects	Severity	Priority	Test case ID	Assigned To	Assigned Date	Status	Resolution Date	Expected Result	Actual Result
DI_1_1	01-09-2023	Tina	Checkout error after payment fails	Enter the payment details correctly	High	High	OSS_3_1	Daniel	01-09-2023	Open	-	Able to pay and checkout	Payment successful but checkout fails
DI_1_2	03-09-2023	Roy	Broken Image	Visit product page or refresh the page	Low	Medium	OSS_2_6	Daniel	04-10-2023	Fixed	05-09-2023	Image displayed	Successful
DI_1_3	05-09-2023	Tina	Login Issue	Login with valid credentials	High	High	OSS_1_1	Nisha	04-10-2023	Closed	05-09-2023	Error message	Successful
DI_1_4	06-09-2023	Nikita	Search product issue	Enter search query properly	Medium	Medium	OSS_2_1	Nisha	07-10-2023	Open	-	Relevant search	Fail
DI_1_5	06-09-2023	Rohan	Add to Wishlist but product added into cart	1.)Select the item and choose the appropriate option 2.)Refresh the page	High	High	OSS_3_1	Neha	07-10-2023	Closed	06-09-2023	Item added to wishlist	Successfull

----

****

Class: SY-MCA   **Shift / Div:** A   **Batch:** S2   **Roll Number:** 52043

Name: Vanessa Reetu Prashant More   **Assignment No:** 2   **Date of Implementation:** 29. 9. 23

****

Q3. Prepare Defect Repository for Mobile Application – Banking App using MS Excel

Project Name:	Vamo Banking App
Module Name:	Authentication, Account transactions & Profile Management
Release:	3.3 version
Program Manager:	Vanessa More
Project Manager:	Ronak Bafna

Defect ID	Date Created	Created By	Defect Description	Steps to reproduce the defects	Severity	Priority	Test case ID	Assigned To	Assigned Date	Status	Resolution Date	Expected Result	Actual Result
DI_2_1	03-09-2023	Kiran	Login Issue	Login with valid credentials	High	High	VBA_1_1	Reetu	03-10-2023	Closed	03-09-2023	Error message	Successful
DI_2_2	05-09-2023	Bhushan	Transaction amount mismatch	Initiate a fund transfer with an amount that matches the intended	Medium	Medium	VBA_2_4	Sneha	06-10-2023	Open	-	Successful transaction	Transaction failed
DI_2_3	07-09-2023	Riya	Account balance display error	Check account credentials properly	Medium	Medium	VBA_2_4	Sneha	07-10-2023	Open	-	Display current account balance	Fails
DI_2_4	07-09-2023	Tina	UI Inconsistency	Navigate to differnet screens shows different font size	Medium	Medium	VBA_2_3	Rohit	08-10-2023	Open	-	Consistent UI	UI elements
DI_2_5	08-09-2023	Ashutosh	Payment Failure	Initiate transaction with sufficient funds	Low	Low	VBA_2_8	Kriti	09-10-2023	Closed	09-10-2023	Successful transaction	Successful

----

****

Class: SY-MCA   **Shift / Div:** A   **Batch:** S2   **Roll Number:** 52043

Name: Vanessa Reetu Prashant More   **Assignment No:** 3   **Date of Implementation:** 3. 11. 23
****

Q1. Write black box test cases using ECP & BVA to test the requirement –

Bank has different charges depending on the transaction done

5% of the amount for transaction less than or equal to 5000

6% of the amount for transaction more than 5000 and less than or equal to 7000

7% of the amount for transaction more than 7000

Name : Vanessa More
SYMCA   Div : A
Roll no : 52043   Batch : S2
Subject : Software Testing Laboratory
Assigment 3.

Q1. Write black box test cases using ECP & BVA to test the requirement –
Bank has different charges depending on the transaction done.
5% of the amount for transaction less than or equal to 5000
6% of the amount for transaction more than 5000 & less than or equal to 7000
7% of the amount for transaction more than 7000

⟶

BVA

Invalid Position	5% Valid Position	6% Valid Position	7% Valid position
0, 2	4999	5001	6999	7001
1	5000		7000	

ECP:

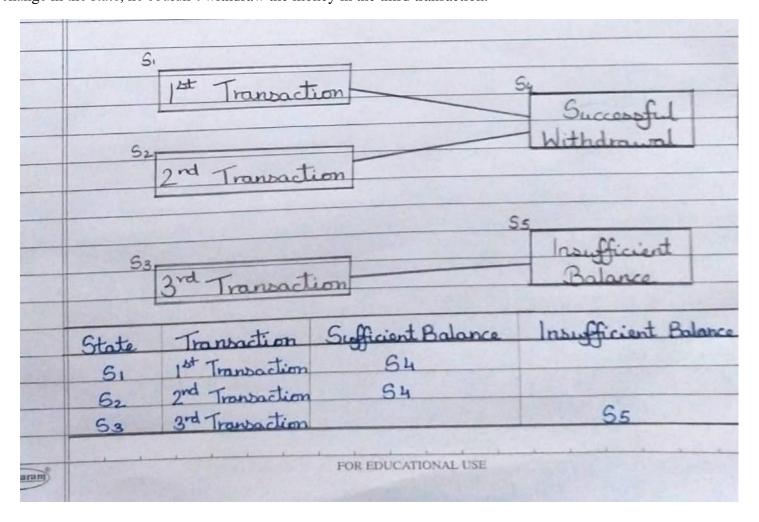| Test Case Id	Partition	Type of input	Test Data	Expected output
BR_1	transaction amt. <=5000	Valid	1, 2, 4999, 5000	Transaction charges=5% of amt.
BR_2	transaction amt. >=5000 & <=7000	Valid	5001, 6999, 7000	Transaction charges=6% of amt.
BR_3	transaction amt. >7000	Valid	7001	Transaction charges=7% of amt.
BR_4	transaction amt. <1	Invalid	0	Error message "Invalid amt."

****

Class: SY-MCA   **Shift / Div:** A   **Batch:** S2   **Roll Number:** 52043

Name: Vanessa Reetu Prashant More   **Assignment No:** 3   **Date of Implementation:** 3. 11. 23

****

Q2. Write black box test cases using state based(state transition) technique to test the requirement –

Withdrawal of money from ATM. 'User A' wants to withdraw 30,000 from ATM. Imagine he could take 10,000 per transaction and total balance available in the account is 25,000. In the first two attempts, he could withdraw money. Whereas in the third attempt, ATM shows a message as "Insufficient balance, contact Bank". Same Action but due to change in the state, he couldn't withdraw the money in the third transaction.

Test Case Id	Test scenario	Test step	Test data	Expected output	Actual output
ATM_1	Check withdraw money transaction amt.	withdraw Rs.10000	10000	Avail. balance=15000	Avail. balance=15000
ATM_2	Check withdraw money transaction amt.	withdraw Rs.10000	10000	Avail. balance=5000	Avail. balance=5000
ATM_3	Check withdraw money transaction amt.	withdraw Rs.10000	10000	Error message:"Insufficient balance "	Error message:"Insufficient balance "

****

Class: SY-MCA   **Shift / Div:** A   **Batch:** S2   **Roll Number:** 52043

Name: Vanessa Reetu Prashant More   **Assignment No:** 3   **Date of Implementation:** 3. 11. 23

****
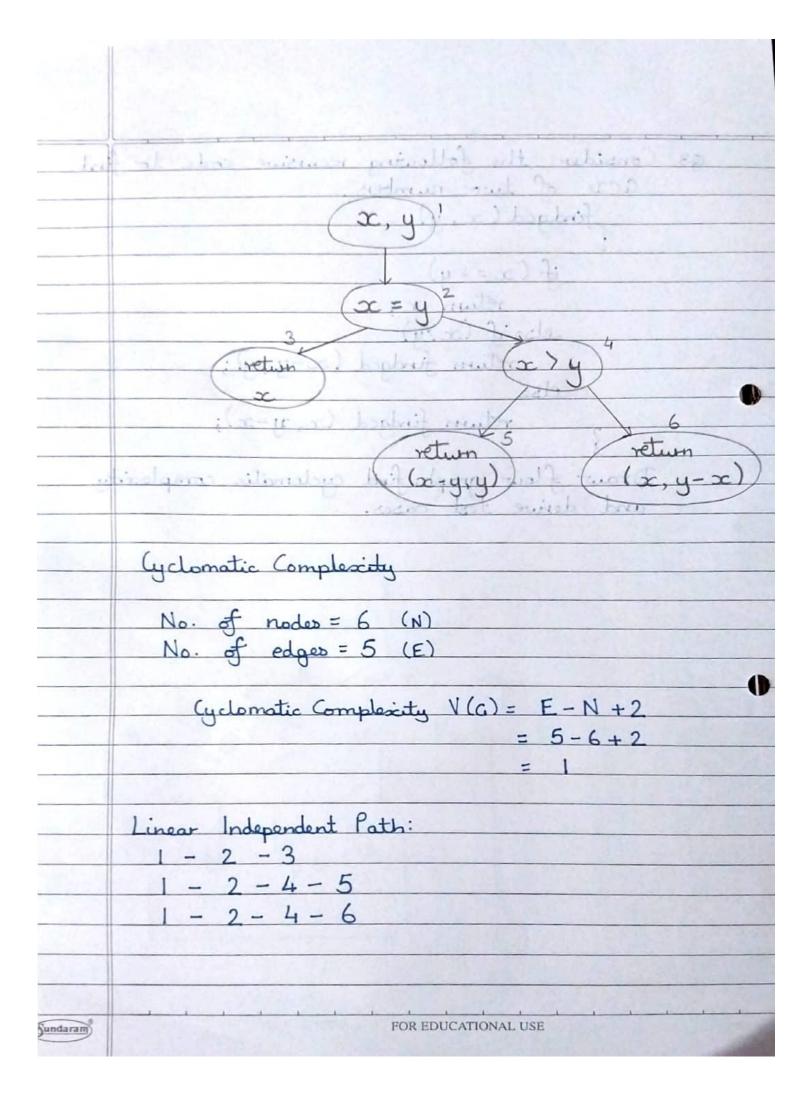
Q3. Consider the following recursive code to find GCD of two numbers

findgcd (x , y)
{
if (x = y)
return x;
else if (x>y)
return findgcd(x-y,y);
else
return findgcd(x,y-x);
}

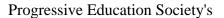Draw flow graph, find cyclomatic complexity and derive test cases

Cyclomatic Complexity

No. of nodes = 6 (N)
No. of edges = 5 (E)

Cyclomatic Complexity $V(G) = E - N + 2$
$$= 5 - 6 + 2$$
$$= 1$$

Linear Independent Path:

1 - 2 - 3
1 - 2 - 4 - 5
1 - 2 - 4 - 6

Test Case Id	Test Case Description	Type of Input	Steps	Test data	Expected output	Actual output	Status
TC_001	find_gcd(x,y)	Valid	Read x and y such that x=y	x=5, y=5	return x	return x	Pass
TC_002	find_gcd(x,y)	Valid	Read x and y such that x>y	x=8, y=4	return find_gcd(x,y-x)	return find_gcd(x-y,y)	Pass
TC_003	find_gcd(x,y)	Valid	Read x and y such that x<y	x=3, y=9	return find_gcd(x-y,y)	return find_gcd(x-y,y)	Pass

----

****

Class: SY-MCA   **Shift / Div:** A   **Batch:** S2   **Roll Number:** 52043

Name: Vanessa Reetu Prashant More   **Assignment No:** 4   **Date of Implementation:** 24. 11. 23

****
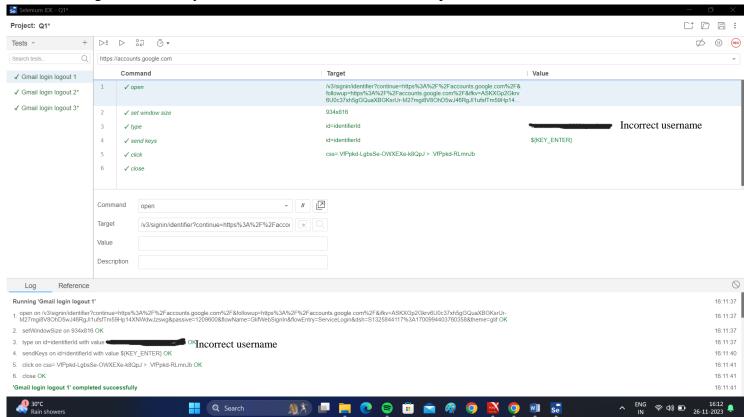
Q1. Install Selenium IDE in Mozilla Firefox/ Google Chrome

Using "https://accounts.google.com" Test Gmail Login Logout functionality in IDE (Record and Replay)

- Test Gmail login functionality with incorrect username and correct password.
- Test Gmail login functionality with incorrect username and incorrect password.
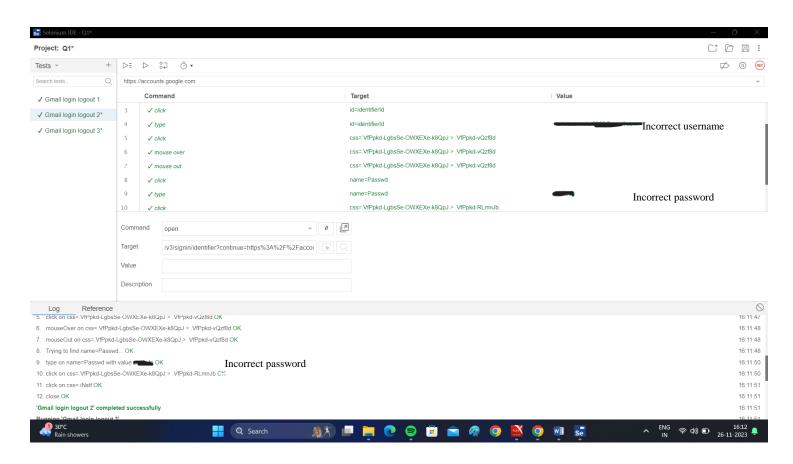- Test Gmail login functionality with correct username and correct password.
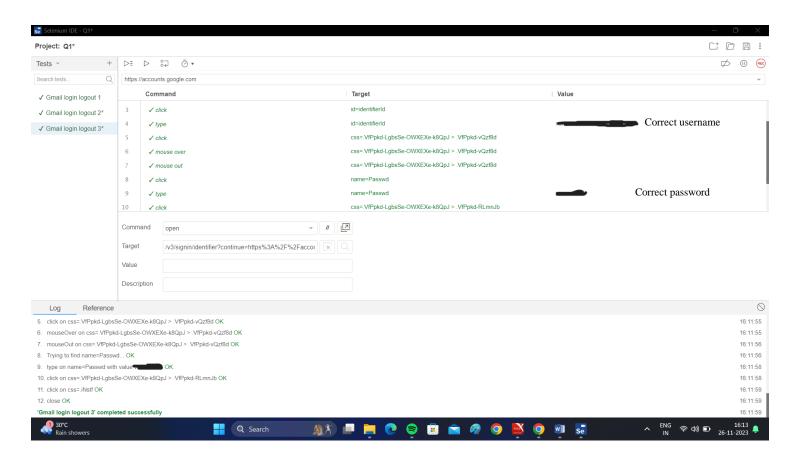

Output:

Test Gmail login functionality with incorrect username and correct password.

Test Gmail login functionality with incorrect username and incorrect password.

Test Gmail login functionality with correct username and correct password.

****

Class: SY-MCA   **Shift / Div:** A   **Batch:** S2   **Roll Number:** 52043

Name: Vanessa Reetu Prashant More   **Assignment No:** 4   **Date of Implementation:** 24. 11. 23

****

Q2. Test following scenario in Selenium IDE using record and replay
Invoke Google Chrome browser.
● Open URL: www.google.com
● Click on the Google Search text box.
● Type the value "java tutorials"
● Click on any of java tutorial link

Output:

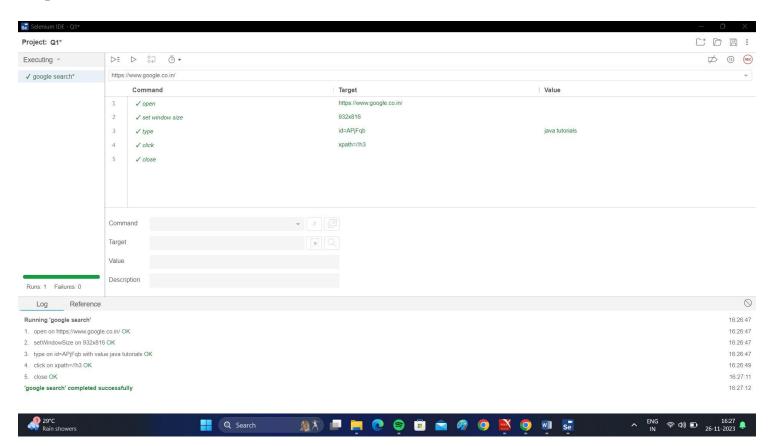**

Class: SY-MCA   **Shift / Div:** A   **Batch:** S2   **Roll Number:** 52043

Name: Vanessa Reetu Prashant More   **Assignment No:** 4   **Date of Implementation:** 24. 11. 23

****

Q.3 Automate Gmail/Rediffmail/Hotmail login flow using Selenium Web Driver

Code:

```
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;


public class login {

        public static void main(String[] args) {
                // TODO Auto-generated method stub
                System.setProperty("webdriver.chrome.driver", "E:\\softwares\\chromedriver-
win64\\chromedriver-win64\\chromedriver.exe");

                WebDriver driver = new ChromeDriver();
                driver.get("https://accounts.google.com");
                driver.findElement(By.id("identifierId"));
                driver.findElement(By.id("identifierId")).sendKeys("***************");
                driver.findElement(By.xpath("//*[@id='identifierNext']")).click();
                driver.findElement(By.name("password")).sendKeys("********");
                driver.findElement(By.xpath("//*[@id='passwordNext']")).click();


        }

}
```

****

Class: SY-MCA   **Shift / Div:** A   **Batch:** S2   **Roll Number:** 52043

Name: Vanessa Reetu Prashant More   **Assignment No:** 4   **Date of Implementation:** 24. 11. 23

****

Q.4. Handle multiple windows using Selenium. (use site naukri.com) using SeleniumWeb Driver.

Code:

```java
import java.util.Iterator;
import java.util.Set;

import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;

public class multipleWindow {

    public static void main(String[] args) throws Exception {
        System.setProperty("webdriver.chrome.driver", "E:\\softwares\\chromedriver-win64\\chromedriver-win64\\chromedriver.exe");

        WebDriver driver = new ChromeDriver();

        driver.manage().window().maximize();
        // To open Naukri.com with multiple windows
        driver.get("https://www.naukri.com/");
        Thread.sleep(2000);
        // It will return the parent window name as a String
        String parentWinID = driver.getWindowHandle();
        System.out.println("Parent Win ID is: " + parentWinID);
        // It returns no. of windows opened by WebDriver and will return Set of Strings
        Set<String> allWinID = driver.getWindowHandles();
        System.out.println("Total Window size:" +allWinID.size());
        System.out.println("All win IDs are:");
        for (String allIDs : allWinID) {
            System.out.println(allIDs);
        }
        // Using Iterator to iterate with in windows
        Iterator<String> itr = allWinID.iterator();
        while (itr.hasNext()) {
            String childWinID = itr.next();
```

// Compare whether the main window is not equal to child window. If not equal, we will close child windows.

```java
                    if(!parentWinID.equalsIgnoreCase(childWinID)) {
                     driver.switchTo().window(childWinID);
                     System.out.println("Child URL is:"+driver.getCurrentUrl());
                     System.out.println("Child Win Title is:"+driver.getTitle());
                     Thread.sleep(2000);
                     driver.close();
                     }
            }
            // This is to switch to the main window
            driver.switchTo().window(parentWinID);
            driver.quit();

        }

    }
```

Output:

```
Parent Win ID is: 0D692EF85935CBBC19273CD7F1913210
Total Window size:1
All win IDs are:
0D692EF85935CBBC19273CD7F1913210
```